

Spécifications

School’ In

Logo

Version 1.0

31/03/2015

Historique des révisions

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Auteur** |
| 21/10/2015 | 1 | 1er regroupement de doc | Victor |
| 27/10/2015 | 2 | 1er correction | Victor |
|  |  |  |  |
|  |  |  |  |

Sommaire

1 Introduction 4

1.1 Mission 4

1.2 Objectifs 4

1.3 Glossaire 4

2 Description générale 5

2.1 Acteurs 5

2.2 Cas d’utilisations 5

3 Spécifications fonctionnelles 6

3.1 Carte de navigation 6

3.2 Détails des cas d’utilisations 6

4 Spécifications non fonctionnelles 9

4.1 Contraintes de conception et d’implémentation 9

4.2 Documentation utilisateur 9

# Introduction

## Mission

Nous allons concevoir et réaliser une application client de gestion d’une école afin de faciliter le travail du service pédagogique. Cette application permettra à un établissement de réaliser des taches tel que l’assignation des professeurs à différentes classes, de définir si un élève passe de classe ou redouble.

## Objectifs

* Faciliter la gestion administrative d’une école (avant le 23/11/2015).
  + Réaliser une interface simple d'utilisation.
* Optimiser le suivi pédagogique des élèves (avant le 14/12/2015).
  + Mettre une place une gestion des absences.
  + Créer une gestion des élèves et des professeurs.
* Simplifier la mise en place des plannings scolaires (avant le 18/01/2016).
  + Créer une gestion des classes et des salles.
  + Réaliser une proposition automatique de planning.

## Glossaire

### Termes du domaine

School'In: Nom du projet.

Trello : Trello est un outil de gestion de projet en ligne.

Intech’ Info : L'école à laquelle appartient les membres de l'équipe School'IN.

### Termes techniques

Pascal case : C'est une convention qui spécifie que la première lettre de chaque mot doit être en majuscule.

Nunit : Nunit est un cadre de travail open source de tests unitaires, pour Microsoft.net

Visual Studio : C'est un ensemble complet d'outils de développement permettant de générer des applications.

# Description générale

## Acteurs

|  |  |
| --- | --- |
| Administrateur | L’administrateur est celui qui fait la gestion de l’appli. C’est lui l’acteur principal, il interagit le plus avec le système, il a pour mission de :   1. Créer une école. 2. Créer une classe. 3. Créer sale. 4. Créer planning. 5. Ajouter élève. 6. Créer planning 7. Gèrer école. 8. Supprimer une classe. 9. Supprimer salle. 10. Ajouter une classe. 11. Supprimer une école.. 12. Modifier planning. 13. Changer salle. 14. Retirer élève. |

## Cas d’utilisations

1.Créer une école.

2.Créer une classe.

3.Créer sale.

4.Créer planning.

5.Ajouter élève.

6.Créer planning

7.Créer des cours

8.Gèrer école.

9.Supprimer une classe.

10.Supprimer salle.

11.Ajouter une classe.

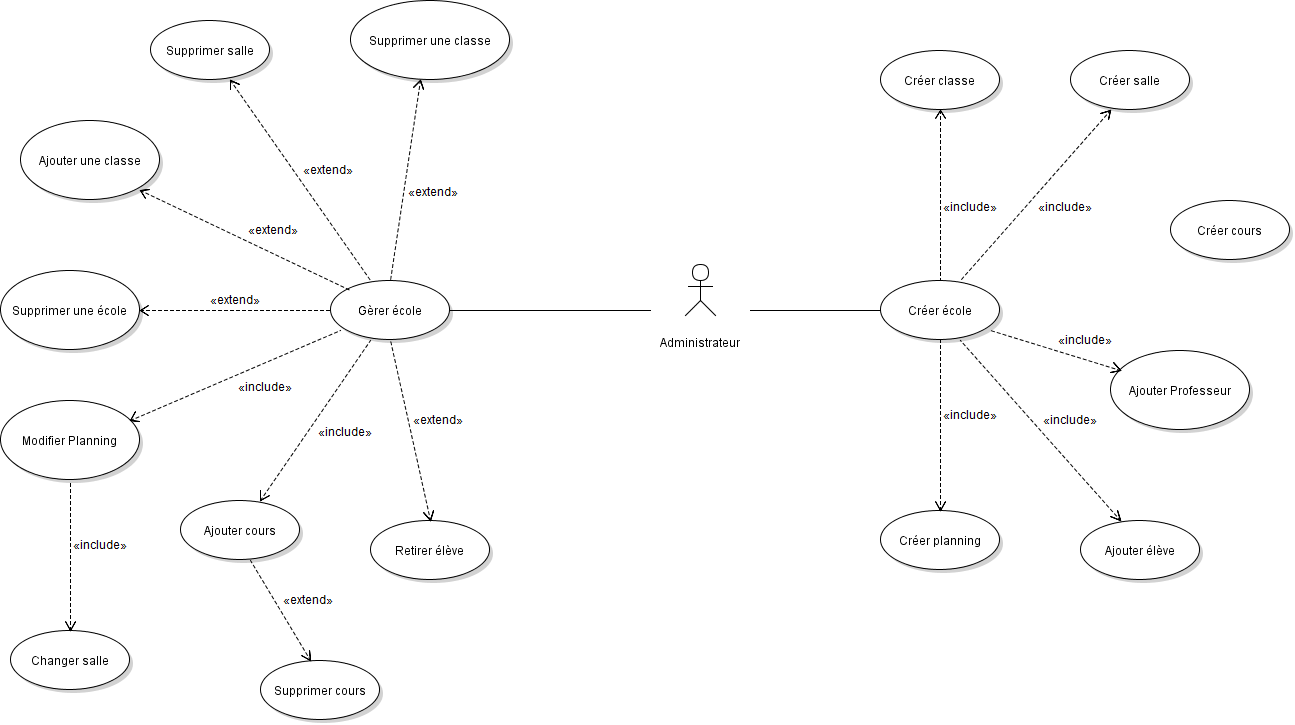
12.Supprimer une école..

13.Modifier planning.

14.Changer salle.

15.Retirer élève.

### Diagramme de cas d’utilisation



### Listes des cas d’utilisations

|  |  |  |  |
| --- | --- | --- | --- |
| **Cas d’utilisations** | **Acteurs** | **Priorité** | **Risques** |
| 1. Créer une école. | Administrateur | Avant de gèrer une école,l’administrateur devra en créer une . | Risque que les informations de l’ancienne école se retrouve dans la nouvelle. |
| 2. Créer une classe. | Administrateur | L’administrateur crée une classe pour pouvoir la gérer | Créer plusieurs classes sans utilities. |
| 3. Créer salle | Administrateur | L’administrateur crée une salle pour pouvoir la gérer | Un peu comme les classes,en créer plusieurs sans s’en server. |
| Créer planning | Administrateur | L’administrateur crée un nouveau planning pour pouvoir le gérer |  |
| Ajouter élève | Administrateur | L’administrateur crée un nouvelle eleve pour pouvoir le gérer |  |
| Gèrer école | Administrateur | L’administrateur modifie les information de l’école |  |
| Supprimer une classe | Administrateur | L’administrateur suprime les informations lier a la classe |  |
| Supprimer une salle | Administrateur | L’administrateur suprime les information lier a la salle |  |
| Ajouter une classse | Administrateur | L’administrateur modifie les informations de la classe |  |
| supprimer une école | Administrateur | L’administrateur suprime toute les information de l’école |  |
| Modifier planning | Administrateur | L’administrateur modifie les informations du plannig |  |
| Changer salle | Administrateur | L’administrateur modifie les informations du plannig |  |
| Retirer élève | Administrateur | L’administrateur suprime toute les information de l’éléve |  |

# Spécifications fonctionnelles

## Carte de navigation

## Détails des cas d’utilisations

L'acteur principal actionne un fichier .exe en cliquant dessus.

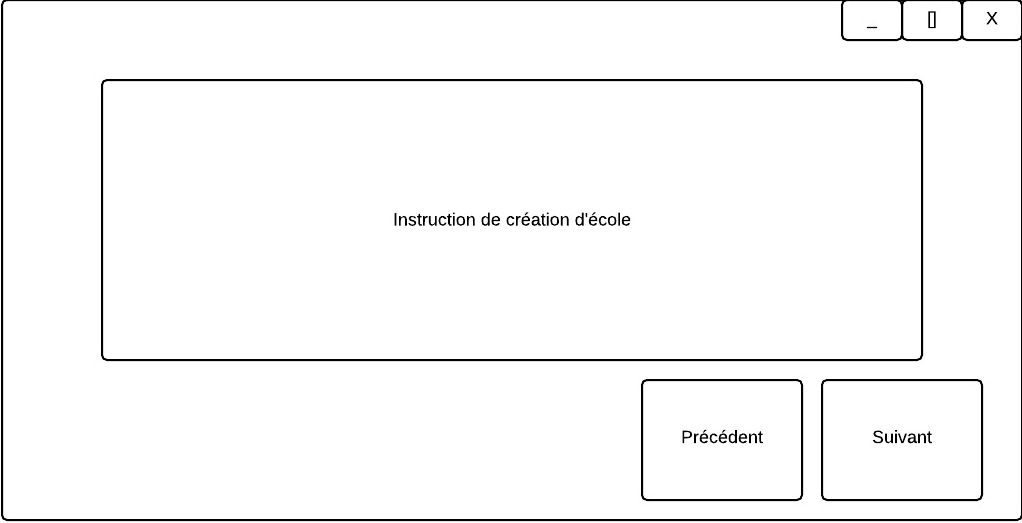
A tout moment l'application peut être arrêtée par le système grâce à un bouton "arrêt" en haut à droite de l'application. Il y a également deux autres boutons en haut à droite de l'écran pour zoomer et dézoomer la fenetre.

A chaque action terminé (ex: création d'une école), le système revient automatiquement à la page d'accueil.



L'application s'ouvre avec deux boutons  :

Créer ou Modifier.



Dans le cas de la création d'une école une fenetre s'ouvre et demande de remplir au fur et à mesure les informations demander nécéssaire à la création d'une école.

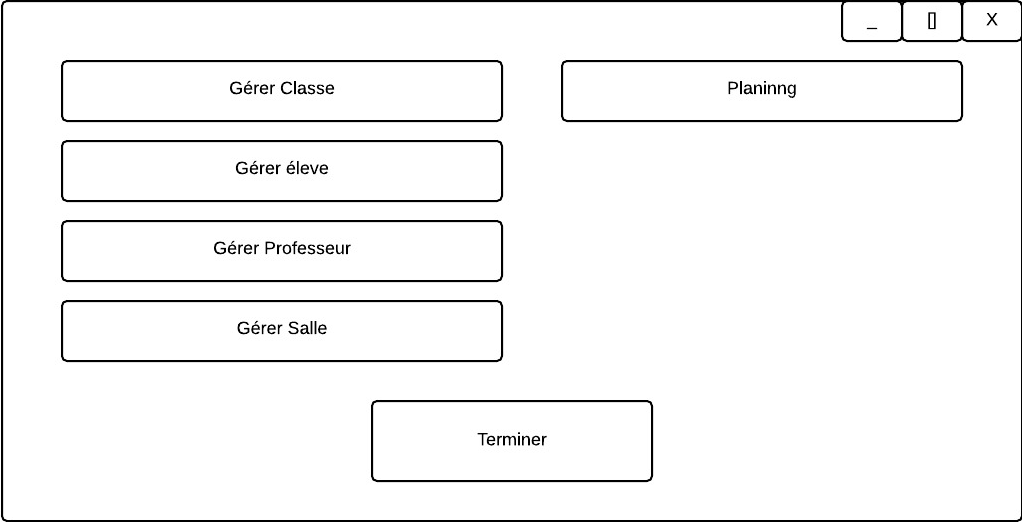
- Information sur l'école (nom, emplacement...)

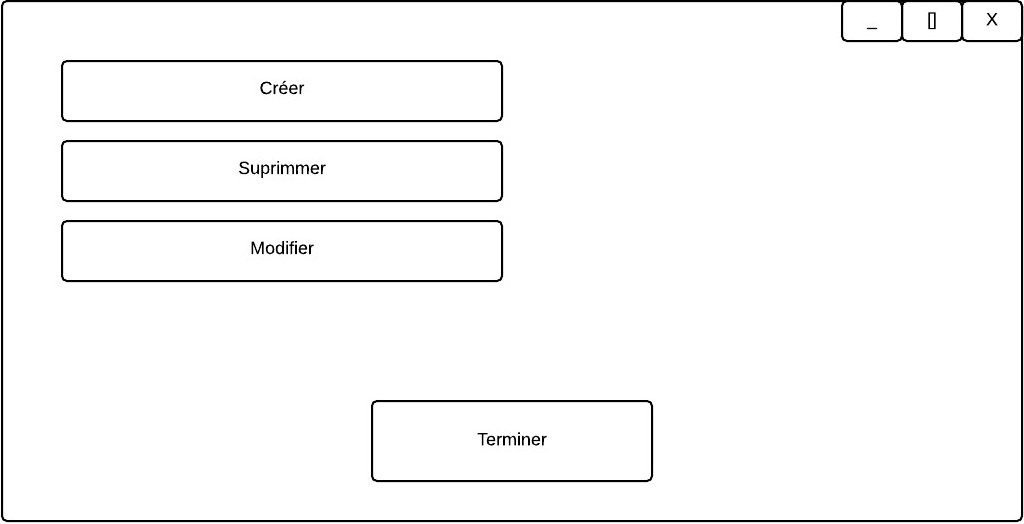
- Eleve associé à l'école  nom,prenom, age...)

- Professeur associé à l'école (nom,prenom,matière enseigné...)

- Classe associé à l'école (nom,nombre d'eleves...)

- Planning associé à l'école





Dans le cas de la modification d'une école une fenetre s'ouvre avec un menu permettant de choisir quel éléments de l'école modifier (élève,professeur,classe,planning).

La modification prend en compte :

Le changement d'information  d'un élément (élève,professeur,classe ou planning).

La suppression d'un élément  (élève,professeur,classe ou planning).

# Spécifications non fonctionnelles

## Contraintes de conception et d’implémentation

Lien pour les conventions de codage : https://msdn.microsoft.com/fr-fr/library/ff926074.aspx

**Contraintes**

- Le langage de programmation utilisé est le c#.

- Les outils utilisés sont Visual studio avec Nunit pour les tests unitaire.

- Un fichier sera utilisé pour stocker les informations relatives à l'application.

- Une documentation sera fournie afin d'expliquer le fonctionnement du programme.

- Les conventions sont indiquées dans le document Conventions.txt

## Documentation utilisateur

Une documentation sera fournie aux utilisateurs afin d'expliquer le fonctionnement de l'application.